EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Jonathon A. Szumny (Reg No. 57,695) on 03/18/2010.

Please Amend the claims as follows:

1-8. (Canceled)

9. (Currently Amended) A computer implemented method comprising:

monitoring usage of a computing resource utilized by a workload wherein said monitoring is performed by a process within a user space and the process monitors the user space only, wherein said process is not an operating system kernel process; wherein the workload includes a plurality of running processes, the plurality of running processes are a subset of all processes that are running in the user space;

accessing a list of memory pages assigned to said workload in the user space; and

responsive to exceeding a limit on utilization of said computing resource, decreasing usage of said computing resource by said workload, ~~said computing resource comprises physical memory and~~ said decreasing usage of said computing resource comprises paging a portion of said ~~physical~~ list of memory pages assigned to said workload out of said ~~physical~~ list of memory pages and said decreasing usage ~~does not halt operation of said workload~~ allows at least partial operation of said workload during said paging such that the at least partial operation does not stop said workload; wherein said portion of said list of memory pages comprises a least recently used portion of said list of memory pages assigned to said workload.

10. (Canceled)

11. (Canceled)

12. (Canceled)

13. (Canceled)

14. (Original) The method of claim 9 wherein said decreasing usage is initiated by a process of said workload.

15. (Canceled)

16. (Currently Amended) A computer implemented method for memory management of a workload from within the workload comprising:

accessing a list of memory pages assigned to said workload in a user space, the workload includes a plurality of running processes, the plurality of running processes are a subset of all processes that are running in the user space;

responsive to a request from a first process of said workload for memory which exceeds a predetermined memory limit for said workload, selecting a plurality of memory pages from said list of memory pages, wherein the plurality of memory pages includes least recently used memory pages assigned to the workload, wherein the plurality of memory pages are least recently used by the workload; and

initiating a second process within the user space to page out said plurality of memory pages, wherein said second process is not an operating system kernel process, wherein at least a portion of said workload continues to operate subsequent to said initiating.

17. (Original) The method of claim 16 wherein said accessing, selecting and initiating are performed by said second process within said workload.

18. (Canceled)

19. The method of claim 16 wherein said second process is loaded into a user space.

20. (Canceled)

21. (Canceled)

22. (Currently Amended) The method of claim ~~20~~ 16 wherein said page out of said plurality of least recently used memory pages reduces a number of memory pages assigned to said workload to below said memory limit.

23. (Currently Amended) The method of claim ~~20~~ 16 wherein said plurality of least recently used memory pages comprises the minimum number of memory pages to reduce said number of memory pages assigned to said workload below said memory limit.

24. (Canceled)

25. (Original) The method of claim 16 wherein said initiating is not performed by an operating system kernel process.

26-35. (Canceled)

36. (Currently Amended) A computer implemented method comprising:

accessing memory usage for a workload and examining page usage for each process of said workload, the workload exists within a user space and includes a plurality of running processes, the plurality of running processes are a subset of all processes that are running in the user space;

aggregating usage of said each process to determine an aggregate usage for said workload;

monitoring said memory usage for the workload by a process within the user space that is not an operating system kernel process;

if said aggregate usage does not exceed a memory utilization limit for said workload, repeating said accessing and aggregating for a next workload;

if said aggregate usage exceeds said memory utilization limit for said workload, determining least recently used pages by the workload based on accessed bits associated with said workload;

if said aggregate usage exceeds said memory utilization limit for said workload, supplying a range of the least recently used pages in a system call to an operating system kernel for evicting said range of the least recently used pages to reduce resource usage by said workload; and retaining at least partial operation of said workload during said page evicting such that the at least partial operation does not stop said workload.

37. (Original) The method of claim 36 wherein said determining and said supplying occur in a plurality of user space processes.

38. (Currently Amended) A computer implemented method of managing computer resources over a plurality of workloads, said method comprising:

for each workload of said plurality of workloads, monitoring respective workload resource usage against a respective allotment of each workload;

determining a range of computer resources to page out for each workload whose resource usage exceeds its respective allotment, wherein said determining comprises determining least recently used pages for each workload whose resource usage exceeds its respective allotment; and

initiating a paging out operation of said range of computer resources and wherein said monitoring, said determining and said initiating all occur within a process of user space that is not an operating system kernel process,

wherein each of the plurality of workloads exists within a user space and includes a plurality of running processes, the plurality of running processes are a subset of all processes that are running in the user space, and paging out said range of computer resources and wherein each workload whose resource usage exceeds its respective allotment remains partially operable during said paging out of its respective range of computer resources.

39. (Canceled)

40. (Original) The method as described in claim 38 wherein said process is situated within a workload of said plurality of workloads.

41-53. (Canceled)

-- END OF AMENDMENT --

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ERIC C. WAI whose telephone number is (571)270-1012. The examiner can normally be reached on Mon-Fri, 9am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng - Ai An can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Meng-Ai An/
Supervisory Patent Examiner, Art Unit 2195

/Eric C Wai/
Examiner, Art Unit 2195